Flexible Text Classification for Financial Applications: The *FACILE* System

Fabio Ciravegna[°] and **Luca Gilardoni**[♣] and **Alberto Lavelli**[°] and **Silvia Mazza**[♣] and **William J. Black**[♠] and **Massimo Ferraro**[♣] and **Nadia Mana**[°] and **Johannes Matiasek**[♦] and **Fabio Rinaldi**[♠]

Abstract. This paper describes an advanced system for multilingual text classification adaptable to different user needs. The system has been initially developed as an applied research project involving both research centres, industrial bodies and end-user organizations. The project is a considerable success story in the financial field. Three different successful applications were released at the users' sites involved in the project. Moreover the system was adopted by the main Italian financial news agency where it is used to provide classified news for an external pay-to-view service. The system has been running continuously since January 1998. Its architecture integrates modules based on both innovative artificial intelligence methodologies and commercial tools. This shows that state-of-the-art AI techniques are mature enough to provide real world applications.

1 PROBLEM DESCRIPTION

Knowledge is nowadays the key source for competitive advantage. The success or failure of a company can depend on the ability to find the right information at the right time. The www explosion (and the increasing usage of Internet technologies as a core channel for communication) multiplies the sources of information and increases by orders of magnitude the amount of information available. However, while raising the opportunities for gaining competitive advantages, this also increases the information glut. The main value is not in the information itself, but in the capability of managing it successfully to derive knowledge that is critical to an organisation’s objectives. Successfully managing information means being able to correctly integrate it with existing structured information, to facilitate communication and knowledge sharing and to support knowledge-based organisations.

The role of natural language processing and artificial intelligence is fundamental in this respect as: i) the vast majority of this information is textual and available in different languages; ii) the development of new tools for structuring textual data starting from its content represents one of the fundamental steps in successfully managing information.

This is particularly evident in the business arena, where on-line textual information from news providers has long since been available and heavily used. Reports from Gartner Group [Bair, 1998] explicitly mention advanced classification systems characterised by semantic technologies as the most strategically relevant element to support effective knowledge management.

The aim of the project described in this paper was the study and implementation of a system for text classification to be used in the business world. During the first phase of the project the kind of classification needed for such applications was studied. Two main characteristics were identified: flexibility and refinability. **Flexibility** is needed with respect to both the number of the categories and the granularity of the classification to be coped with; we discriminate among three main types of classification: coarse-grained, fine-grained, and content-based. *Coarse-grained* classification is performed among a relatively small number of classes (e.g., some dozens) that are sharply different (e.g., sport vs finance). *Fine-grained* classification is performed over a usually larger number of classes that can be very similar (e.g., discriminating between news about private bond issues and news about public bond issues). Sometimes categories are so similar that classification needs to be *content-based*, i.e. it can be performed only by extracting the news content (e.g., finding news articles issued by Italian financial institutions referring to amounts in excess of 100,000 Euro). Performing such classification is equivalent to allowing a user to make queries about (part of) the text content.

Refinability was defined as the possibility of performing classification in a sequence of steps, each one providing a more precise classification (from coarse-grained to content-based). In the current technological situation coarse-grained classification can be performed quickly (e.g., via information retrieval systems), while the systems available for more fine-grained classification (e.g., information extraction systems for content-based classification) are much slower and less general purpose. When the amount of textual material is large an incremental approach, based on some level of coarse-grained classification further refined by successive analysis, proves to be very effective. A refinable classification is generally performed over a hierarchy of classes. A refinement may revise the categories assigned to specific texts with more specialised classes from the hierarchy. More complex techniques are invoked only when needed and, in any case, within an already detected context. Classification over a hierarchy also allows the selection of different classification granularities for different users. A person interested in a specific sector will see just coarse-grained classes for the uninteresting parts of the hierarchy, while seeing fuller details for the interesting ones.

[°] ITC-irst, Centro per la Ricerca Scientifica e Tecnologica, via Sommarive 18, 38050, Povo (TN), Italy

[♣] Quinary SpA, via Fara 35, 20124 Milano, Italy

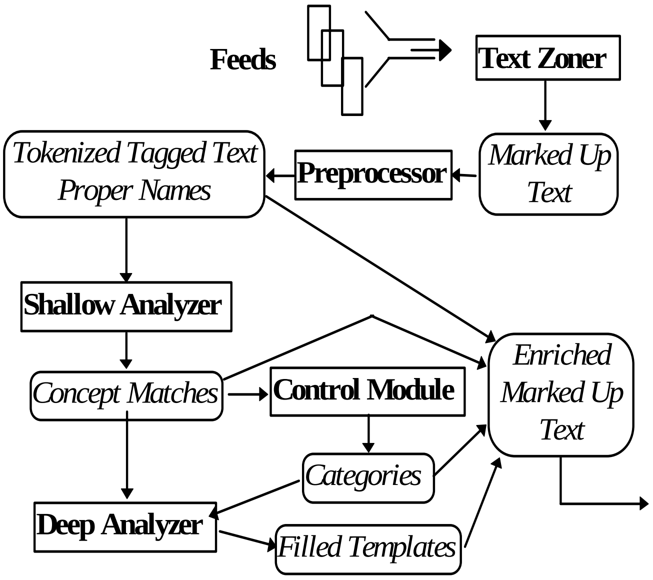
[♠] UMIST - PO Box 88, Manchester M60 1QD, United Kingdom

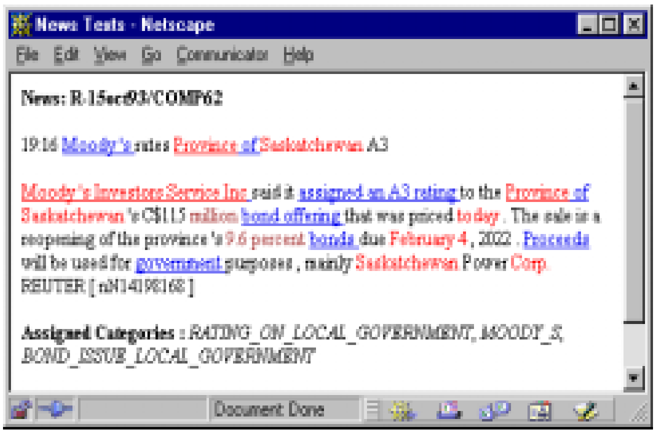
[♦] ÖFAI, Schottengasse 3, 1010 Vienna, Austria

2 APPLICATION DESCRIPTION

The aim of the project was to build a text classification system able to provide flexible and accurate classification for the business world in four languages (Italian, English, German and Spanish).
Much technology currently available on the market still resorts to information retrieval-derived systems and techniques (IR). This technology does not provide adequate accuracy when coping with rich and complex classification structures. This is mainly because IR systems are not able to take into account the linguistic context in which information is inserted. More knowledge intensive methodologies (such as Pattern Matching, henceforth PM) can provide more fine-grained classification, as they are able to take into account part of the linguistic context. PM can produce a fairly accurate and fast categorisation over a large number of classes [Hayes and Weinstein, 1990; Jacobs and Rau, 1990]. Resource development does not require linguistic expertise and can be done by trained users. But PM is still weak on the analysis of linguistic structures and cannot be used for content-based categorisation. Text classification can be performed also using Information Extraction (IE) techniques. IE systems perform very well in detecting texts relevant for single classes (e.g., management succession). Unfortunately IE cannot be performed on a large number of classes: it is an expensive technology as it requires a large amount of time of linguistically aware personnel [Grishman, 1997]. Moreover IE systems are often not efficient enough to cope with large amount of texts.

The system presented in this paper integrates PM and IE to achieve a classification methodology highly adaptable to different user needs. Pattern-based (shallow) classification performs broad-band text filtering (i.e., coarse and fine-grained classification), comparable to that done by a human quickly skimming texts, i.e. recognising the main topics without careful reading. IE is used to provide content-based classification, comparable to that produced by a careful human reading. IE is used to cope with only some of the classes, i.e. those on which shallow classification results need to be refined, according to the user needs. Classification obtained in this way provides flexibility in different directions: coarse/fine-grained/content-based classification, large/restricted number of classes, high/low efficiency in the process. It is also flexible in terms of application development. The development of PM rules takes less time than that of IE resources: a PM-based application can be provided as a first (although not complete) answer to the user needs, waiting for a more complete integrated PM+IE application to be developed.





News: R 15oct93/COMP62

19:16 Moody's rates Province of Saskatchewan A3

Moody's Investors Service Inc said it assigned an A3 rating to the Province of Saskatchewan 's C$115 million bond offering that was priced today . The sale is a reopening of the province 's 9.6 percent bonds due February 4 , 2022 . Proceeds will be used for government purposes , mainly Saskatchewan Power Corp. REUTER [nN14198168]

Assigned Categories : *RATING_ON_LOCAL_GOVERNMENT, MOODY_S, BOND_ISSUE_LOCAL_GOVERNMENT*

2.1 Architecture

The FACILE system is able to categorise in detail business texts such as agency news, newspaper articles and analysis reports from different sources and written in different languages. Classification is obtained via a multi-step process, where each step adds information to the input. A control module activates the different modules in order to assign the appropriate level of classification needed by the user for each text. To do that, the control module uses the information progressively added by the different modules in the architecture to each text. In order to simplify the integration of the classifier into the user environment, the system architecture is divided in two main blocks: the Application Layer and the Classification Kernel. The **Application Layer** provides interface between the classification kernel and the user environment. It converts the input texts into the classification kernel internal format. It also converts the classification results into the user environment format. The **Classification Kernel** is concerned only with text analysis, expecting as input a normalised, tagged text and providing as output an annotated text. It is currently deployed as a standalone module, configurable within a full development environment or stripped down to a runtime executable. It is accessible via application layers using different communication protocols, (e.g., sockets and http). The classification kernel is composed of the preprocessor, the shallow analyser, the deep analyser. These modules are described in the rest of this section.

2.2 Preprocessor

From the point of view of flexible classification **the preprocessor** implements tasks that are to a large degree application independent, such as segmentation, normalisation of numbers, dates and abbreviations, as well as morphological analysis, part of speech tagging, preliminary unknown word guessing and named entity recognition. It does not require major adaptation for new applications. The preprocessor operates on a zoned text, i.e. a text with different parts (title, body, etc.) already identified and marked up. It provides morpho-syntactic information and a disambiguated syntactic tag for each token in the input, identifying proper names as single tokens and assigning them a (disambiguated) semantic tag. The key steps are the same for all the languages considered (currently English, Italian, German and Spanish) and use the same machinery with language-specific resources. Morphological analysis and part of speech tagging integrate generic resources and commercial machinery (the LinguistX tools produced by InXight^{tm})[1]. Database lookup adds semantic features (e.g., person, organisation, location, etc) for (possibly multi-word) tokens in the database. NEA recognises complete names as well as numbers and time expressions [Black *et al*., 1998].

2.3 Shallow Classifier Module

The shallow classifier (SCM) performs a coarse and fine-grained classification by exploiting techniques that can be easily tailored to new applications by trained personnel. It is based on pattern matching and assigns a set of classes to each text. It operates in two stages: first it recognises domain relevant concepts mentioned in texts, and then assigns categories to the text. The first step is performed by the **shallow analyser,** based on a refined version of the model presented in [Gilardoni *et al*., 1994]. The shallow analyser identifies both domain objects (e.g., shares, market sectors, balance sheet items) and domain events (e.g., joint ventures, public offers, financial transactions) in the input via a set of patterns. The pattern language is similar in spirit to those implemented by search engines, with additional features exploiting information from the preprocessing stage. Patterns can exploit some relation (e.g., inheritance) contained in a knowledge base (KB). Concept patterns are largely language dependent, but the KB is shared across languages. The result of shallow analysis is a set of identified concept references in text with associated confidence factors. The second step is performed by a rule-based **categorizer**. The text's main topic is determined using application-specific heuristics (based on both concept matches and the linguistic context), as well as domain knowledge. The whole machinery is supported by modular declarative resources to describe concepts, patterns, categories and classification rules.

2.4 Deep Analyser Module

The deep analyser (DAM) further refines the classification provided by the SCM. It performs content-based classification via information extraction. DAM's task is similar to the Scenario Template task in MUC conferences [MUC, 1998]. DAM receives as input the pre-processed and classified text and fills relevant templates. Templates are associated to classes in the hierarchy of classification. A sufficiently filled template brings to class assignment. The extracted information allows users to express classification queries such as «send me any text concerning bonds issued by European-based financial institutions whose amount exceeds 1 million Euro».

DAM's architecture is based on a sequence of submodules performing full parsing, lexical semantics, default reasoning, discourse processing and template filling and merging [Ciravegna *et al*., 2000]. The parser produces for each sentence a complete parse tree (or its Finite State Approximation [Ciravegna and Lavelli, 1999]). A Quasi Logical Form (QLF) is produced by a lexical semantic submodule by exploiting the information provided by both the lexicon and the relations established by the parser. Default reasoning is then applied to introduce in the QLF additional information not explicitly contained in the text, but needed for template filling (e.g., «if a person, working for company X, is hired by company Y, s/he is no longer employee of X»). After this, discourse processing is performed: (pro)nominal references are resolved for both objects (people, organisations, physical objects) and events (e.g., hiring/firing); implicit relations are also captured (e.g., «The Bank of Japan decided .... The **president** said ....»). Templates are finally filled by using the final QLF (as produced by an additional default reasoning step). Template merging and recovery actions cope with missing information.

Each submodule in DAM is implemented as a cascade of Finite State Transducers (FSTs). The whole DAM architecture uses the same formalism, rule engine and even the same set of primitives for all the submodules. Portability was a main requirement for DAM. It can be ported to new domains and languages by just modifying declarative resources: the lexicon, the knowledge base and the FST grammars. Uniformity in the formalism for rule definition allows new applications to be developed by a single person. Porting to new applications required one to three man months, depending on the application. Porting to new languages from 4 to 6 months. DAM resources were developed for two languages (English and Italian).

2.5 Implementation

The modules in the FACILE Classification Kernel have been developed in Common Lisp. The Classification Kernel acts as a Unix server and can be smoothly integrated with clients running on different hardware and software architectures (e.g., PCs). It can be stripped down to a runtime executable, accessible by application layers using different communication protocols, including sockets and http. As mentioned, the LinguistX InXight tools have been integrated within the system. Moreover, as we will see in section 3, it has been interfaced with user environments based on a number of commercial tools.

3 APPLICATION BUILDING

The applications described in this paper have been partly developed within the framework of LE-FACILE, an EC sponsored RTD project, which had the twofold aim to advance state of the art in classification and information extraction techniques and to prove adequateness of the approach for industrial applications. The full project was carried on by a Consortium of eight partners, mixing up three research institutions, providing strong background linguistic expertise, two industrial partners (one with a long track in building AI applications), and three user organisations. The project was structured along two parallel tracks. The first concerned the development of the general architecture suitable for building up different applications, with its kernel tools and the core linguistic resources (see Section 2). At the same time a significant effort was devoted in the first year of the project to collecting user requirements, via a strict interaction between industrial partners and end user organisations. Available technology, linguistic resources and the market situation were carefully assessed.

Both kernel and application development started at the end of year 1, partially exploiting results derived from pre-existing experiences [Gilardoni *et al*., 1994]. In particular applications were

[1] InXight^{tm} was chosen as it covers many different languages (and in particular the four coped with in the project) with a single commercial tool. This uniformity allowed an easy integration in the system.

developed for users within the original Consortium, spanning four languages and two domains. Prototypes including developed functionality were released as soon as available to application developers in order to be tested. This continuous development/validation/feed-back approach enabled development teams to stay focused on application needs, ensuring both requirements satisfaction and industrial quality of software.

The first development stage ended at the beginning of year 3 followed by a final version at the end of the third year. Already during winter of year 2, early prototypes allowed to support both the participation in the MUC7 conference and a first deployment of an installation at an external site.

In the rest of this section we focus on two of the three applications developed within the RTD project and on a fourth application built outside the framework of the original project with another user organisation.

The two applications used as a testbed during development were radically different in scope as well in IT environment, even if they shared a common interest for financial information.

The first one was developed with a small Italian rating company. It aimed at classifying different kinds of texts, namely agency news, newspaper articles, Web papers and internal documents, in Italian and English. The classification hierarchy was composed of about 200 categories. Among such categories: company news, with special attention to results (financial, economic and industrial), to events (e.g., joint ventures, new developments, new products), and to sector news (e.g., economic/financial behaviour of market sectors, economic and financial indicators such as demand, supply, import, etc.). Target users are a small set of financial analysts with compelling needs to assess the situation of companies and markets via information gathering. Specific attention has been put on financial indicators, and on exploiting IE techniques for filtering information concerning ratings and bond issues.

The second testbed application has been put in place for a German information broker hosting a Web news page. In this case, the sources are only web pages, collected through a conventional spider. The FACILE system is used to complement a simpler keyword based classification. It is used to provide richer classification in the specific area of economic/financial indicators, mainly related to countries (e.g., the inflation rate, the gross domestic product, etc.), but also to market sectors (e.g., export and import price indexes).

In either case FACILE technology integrates and enhance commercial tools, providing classification and filtering features which could not be supported by conventional technology.

In both applications, the classification kernel run on a Sparc Solaris server, being fed at the Italian rating company with texts coming from news wires, the net and the local repository, and at the German broker site from a web spider. In the former case, texts, once classified, are saved in a Fulcrum based repository, with PC clients able to retrieve results by combination of Boolean queries over categorisation results intermixed with the full text search capabilities of Fulcrum. In the latter case, results are saved in SOIF format on a LINUX system directly supporting the web site. In the two applications above[2], the classification system runs as a black-box server, with end users just enjoying the result of the classification. Development of the applications involved on one side coding domain specific classification knowledge and strategies, and on the other side integration of the server within the IT infrastructure supporting text feeding. The latter was mainly a software integration matter eased by a clean separation between the classification server and the interface layers. The former was a definitely more relevant activity of knowledge and language engineering.

A major testbed for the technology developed has been the fourth application developed. In this case the customer was the major Italian financial news agency (Radiocor), an organisation not involved in the RTD project. FACILE was adopted in order to support automatic classification of their own news (in Italian only). Classified news feed several pay-to-view services provided on the web and at selected customers' sites through intranet deployment. The domain coverage is similar to that described for the Italian rating company, from which most of the domain specific knowledge has been derived. Extensions concerned the coverage of political events, labour market, macro-economy and money markets. The structure of the classification hierarchy as well as most of the classification criteria are however fairly different, as they reflect the different target of generic users rather than of financial analysts. In addition, some information derived by the system, such as identified known companies, are used to hyperlink structured data (e.g., stock data charts, company balance sheets) hosted by the agency site.

The system is currently running completely unattended in real time 24x7 since beginning of 1998. The feeding layer takes texts from the editorial system and provides results to the Oracle repository, backing both the web site and the satellite communication (which were already in place). Its implementation took a few days. The first version of the application went live on the web in less than a month, with enhanced versions provided during the following six months. Such rapid delivery was made possible by the reuse of large parts of the knowledge base developed within the RTD project testbeds, which was eased by the classifier architecture chosen. The kernel software showed to be general enough: it required no modification. Preprocessor resources required almost no change, and the concept recognition stage supported by the existing domain models required only the extensions for previously uncovered parts. Major changes were related to the different classification structure (including a different sectors segmentation), embodied in the categoriser rules. As such rules however mostly reflect domain and contextual strategies based on evidence for concepts mentioned in texts, they can be easily tackled by an experienced knowledge engineer working together with the editorial staff.

4 APPLICATION BENEFITS

The FACILE project started with the aim of providing a very specialised tool, able to support fine-grained categorisation and information extraction capabilities. It had a very specific main target (highly factual business texts) but at the same time with the widest scope. The architecture was envisioned to support multiple languages and to enable exploitation for potentially different applications. The whole RTD project structure, starting from the Consortium composition, was built to reach these aims. In this respect, the overall project surely reached its goal. Four different applications spanning four languages have been successfully developed out of common technology and shared basic linguistic resources and are in use, two of them supporting public information portals. Each of the applications taken alone has still a relatively narrow scope – only one is dealing with two languages at a time over the same conceptual structure, one other only provides integration between structured and unstructured data – but all enabled to solve the concrete specific requirements they were designed for. Quoting the marketing manager of the news agency: «Our users can hyper-navigate concepts, allowing each story to be read from different perspectives. The FACILE system

[2] And also in the third application developed for a Spanish bank and not described in this paper.

facilitates our profiling capability, enabling us to prepare and manage personalised views so users can expand and focus on a topic.» Richness in classification and entity recognition capabilities enabled delivery of service simply not reachable with other approaches. The attained accuracy enabled running unattended services, reaching up to 95% for simpler feature extraction and 90% for classification [Ciravegna *et al*., 1999]. Moreover, other than represent concrete savings in terms of human effort, automatic classification carries over a major benefit in uniformity of judgements, allowing integration of heterogeneous material coming from different sources within a common conceptual structure, easing retrieval and sharing of textual information and integration with pre-existing material.

5 CONCLUSION

We have described a successful system for text categorisation, successfully applied in several applications in four languages for the financial world. The system is highly modular, both in its processing filters and in its declarative resources. The architecture brings together PM and IE. All the modules in the Classification Kernel have been indipendently evaluated by using evaluation tools (see [Ciravegna *et al*., 1999] for details about the evaluation criteria and the results of the experiments). The system has been a considerable success story: it has been adopted by the main Italian financial news agency for providing a pay-to-view internet service. It has been running continuously in the user environment since January 1998. Applications at user sites involved in the development consortium are also available. We contend that this shows that state-of-the-art AI techniques are mature enough to provide real world applications. The architecture is designed in order to separate knowledge intensive resources (DAM's), from less knowledge intensive modules (SCM's), from domain independent modules (preprocessor's). This approach simplifies resource development both in terms of required expertise and in terms of task complexity. In the future we will continue working in the direction of the time-to-market reduction, in order to considerably enlarge the market for fine-grained classification systems. In particular we will investigate the use of machine learning techniques for resource development.

ACKNOWLEDGEMENTS

This work was partially funded by the European Union in the framework of the Language Engineering Sector, project FACILE (LE 2440). ÖFAI is supported by the Austrian BMWV.
Participants to the FACILE project were: Quinary SpA, ITC-irst, Italrating SpA (IT), ÖFAI, Sema SAE (ES), Caja de Segovia (ES), Sis (DE), UMIST-CCL. The authors would like to thank all the project team, in particular P. Prunotto, V. Fiorentino and K. Schirmer.

REFERENCES

[Bair, 1998] J. Bair. *Dimensions of KM Technology Selection*, GartnerGroup publishing, October 1998.

[Black *et al*., 1998] W. J. Black, F. Rinaldi, and D. Mowatt. FACILE: Description of the NE System Used for MUC-7. In [MUC, 1998].

[Ciravegna and Lavelli, 1999] F. Ciravegna and A. Lavelli. Full Text Parsing using Cascades of Rules: an Information Extraction Perspective. In *Proceedings of the 9th Conference of the European Chapter of the Association for Computational Linguistics*, Bergen, Norway, 1999.

[Ciravegna *et al*., 1999] F. Ciravegna, A. Lavelli, N. Mana, J. Matiasek, L. Gilardoni, S. Mazza, M. Ferraro, W. J. Black F. Rinaldi, and D. Mowatt. FACILE: Classifying Texts Integrating Pattern Matching and Information Extraction. In *Proceedings of the 16th International Joint Conference On Artificial Intelligence (IJCAI99)*, Stockholm, Sweden, 1999.

[Ciravegna *et al*., 2000] F. Ciravegna, A. Lavelli, and G. Satta. Bringing Information Extraction out of the Labs: the *Pinocchio* Environment. In W. Horn (ed.), ECAI 2000, *Proceedings of the 14th European Conerence On Artificial Intelligence*, IOS press, Amsterdam, August, 2000.

[Gilardoni *et al*., 1994] L. Gilardoni, P. Prunotto, and G. Rocca, Hierarchical Pattern Matching for Knowledge Based News Categorization. In *RIAO-94, Intelligent Multimedia Information Retrieval Systems and Management*, pages 67-82, New York, October, 1994.

[Grishman, 1997] R. Grishman. Information Extraction: Techniques and Challenges. In: M. T. Pazienza (ed.). *Information Extraction: a multidisciplinary approach to an emerging technology*. Springer-Verlag, 1997.

[Hayes and Weinstein, 1990] P. J. Hayes and S. P. Weinstein. Construe/TIS: A System for Content Based Indexing of a DataBase of News Stories. In *Proceedings of the 2nd Conference of Innovative Applications of Artificial Intelligence*, 1990.

[Jacobs and Rau, 1990] P. S. Jacobs and L. F. Rau. SCISOR: Extracting Information from On-line News. *Communications of the ACM*, 33(11), 1990.

[MUC, 1998] *Seventh Message Understanding Conference Proceedings (MUC-7)*. Available at: `http://www.muc.saic.com/`.